

PENGUNAAN FOR LOOPING

- Penggunaan index ganjil atau genap

```
</head>
<body>
  <h1>All Students</h1>

  <div th:each="student, iterationStatus: ${students}" th:clas="{iterationStatus..odd}? 'odd'">
    <h3 th:text="'No. ' + ${iterationStatus.count}">No. 1</h3>
    <h3 th:text="'NPM = ' + ${student.npm}">Student NPM</h3>
    <h3 th:text="'Name = ' + ${student.name}">Student Name</h3>
    <h3 th:text="'GPA = ' + ${student.gpa}">Student GPA</h3>

    <h3>Kuliah yang diambil</h3>
    <ul th:each="course, iterationStatus: ${student.courses}">
      <li th:text="${course.name} + '-' + ${course.credits} + ' sks'" >
        Nama kuliah-X SKS
      </li>
    </ul>

    <a th:href="'/student/delete/' + ${student.npm}" > Delete Data </a><br/>
    <a th:href="'/student/update/' + ${student.npm}" > Update Data </a><br/>
  </div>
</body>
```

Pertanyaan 1 : Apakah value yang dihasilkan dari \${iterationStatus.odd}?

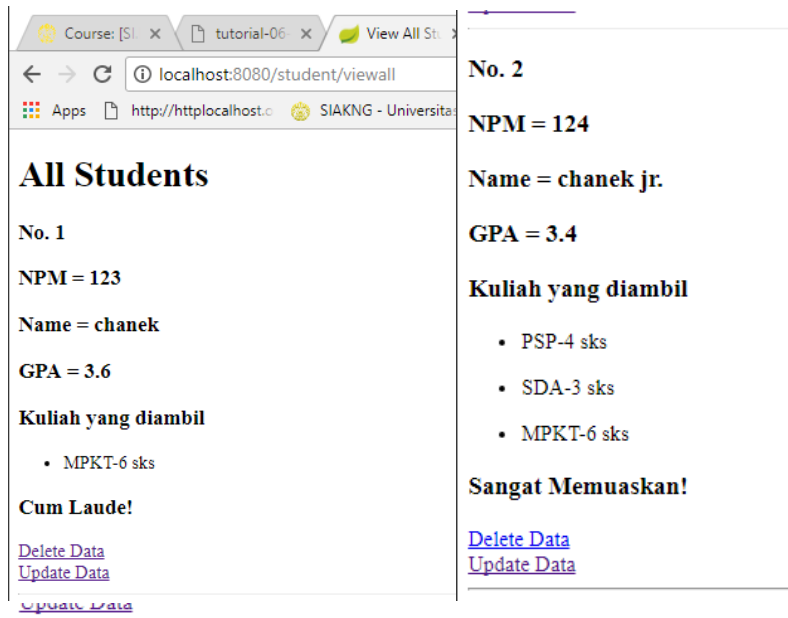
Jawab : value yang dihasilkan merupakan angka – angka ganjil

PENGUNAAN CONDITIONAL EXPRESSION PADA THYMELEAF

1. Tambahkan kode baris berikut ini pada file viewall.html

```
<h3 th:if="${student.gpa}>3.49}">Cum Laude!</h3>
<h3 th:unless="${student.gpa}>3.49}">Sangat Memuaskan!</h3>
```

2. Coba run aplikasi Anda, lalu akses localhost:8080/student/viewall.



The screenshot shows a web browser window with the address bar displaying 'localhost:8080/student/viewall'. The page content includes a header 'All Students' and a list of student profiles. The first profile is for 'chanek jr.' with NPM 123, GPA 3.6, and courses 'MPKT-6 sks'. The second profile is for 'chanek jr.' with NPM 124, GPA 3.4, and courses 'PSP-4 sks', 'SDA-3 sks', and 'MPKT-6 sks'. Both profiles include a 'Cum Laude!' status and links for 'Delete Data' and 'Update Data'.

No. 1
NPM = 123
Name = chanek
GPA = 3.6
Kuliah yang diambil

- MPKT-6 sks

Cum Laude!
[Delete Data](#)
[Update Data](#)

No. 2
NPM = 124
Name = chanek jr.
GPA = 3.4
Kuliah yang diambil

- PSP-4 sks
- SDA-3 sks
- MPKT-6 sks

Sangat Memuaskan!
[Delete Data](#)
[Update Data](#)

No. 3
NPM = 125
Name = sav
GPA = 3.8
Kuliah yang diambil

- PSP-4 sks
- DDP 1-4 sks

Cum Laude!
[Delete Data](#)
[Update Data](#)

Pertanyaan 2 : Mengapa condition di dalam th:unless sama dengan di dalam th:if? Jelaskan jawaban Anda.

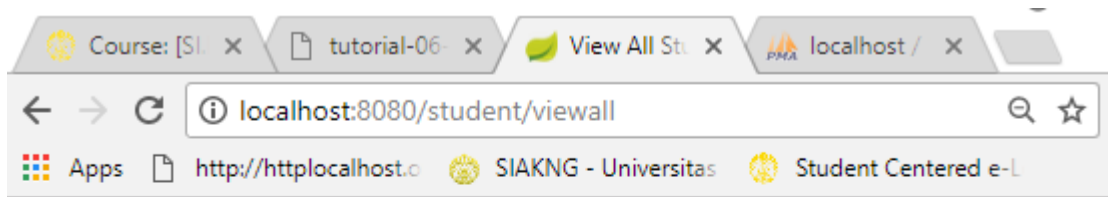
Jawab : Karena th:unless sama saja seperti 'else' atau 'if not' sehingga yang dimaksud th:unless disitu adalah jika gpa student tidak lebih dari atau sama dengan 3.49 maka mendapatkan 'Sangat Memuaskan'.

Pertanyaan 3 : Coba ubah condition pada th:unless=\${student.gpa>=3.49} menjadi th:unless=\${student.gpa<=3.48},

```
<h3 th:if="${student.gpa}>=3.49}">Cum Laude!</h3>  
<h3 th:unless="${student.gpa}<=3.48}">Sangat Memuaskan!</h3>
```

lalu run kembali aplikasi Anda dan mengakses halaman yang sama. Apakah terjadi error? Jika iya, kenapa? Jelaskan alasan Anda. Jika tidak error, apakah hasilnya sesuai?

Jawab : terjadi error yang mana menunjukan pada viewall line 23 yaitu bagian th:unless. Terjadi error karena outputnya sama dengan th:if atasnya dan condition dari keduanya juga sama.



Whitelabel Error Page

This application has no explicit mapping for /error, so you are seeing this as a fallback.

Sun Oct 15 10:55:21 ICT 2017

There was an unexpected error (type=Internal Server Error, status=500).

Exception parsing document: template="viewall", line 23 - column 38

Pertanyaan 4 : Tuliskan cara lain Anda dengan penulisan conditional expression yang berbeda untuk mendapatkan hasil yang sama seperti gambar di atas. Hint: Gunakan ternary operator.

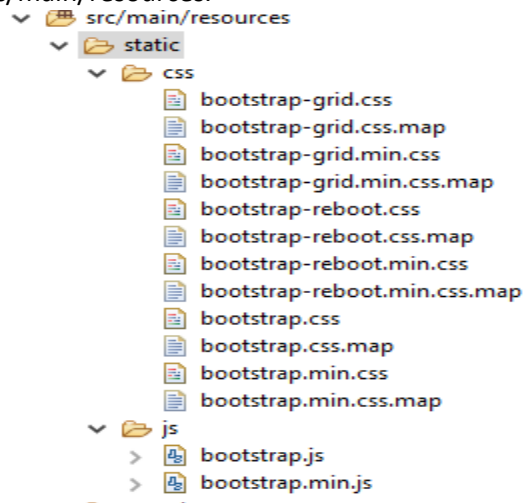
Jawab :

```
<h3 th:text="{student.gpa}>=3.49} ? 'Cum Laude!' : 'Sangat Memuaskan!'"></h3>
```

<div><div>localhost:8080/student/viewall</div><div>Apps http://httplocalhost.c SIAKNG - Universitas Stu</div><div><h2>All Students</h2><div><div>No. 1</div><div>NPM = 123</div><div>Name = chanek</div><div>GPA = 3.6</div><div>Kuliah yang diambil<ul style="list-style-type: none">MPKT-6 sks</div><div>Cum Laude!</div><div>Delete Data Update Data</div></div><div><div>No. 2</div><div>NPM = 124</div><div>Name = chanek jr.</div><div>GPA = 3.4</div><div>Kuliah yang diambil<ul style="list-style-type: none">PSP-4 sksSDA-3 sks</div></div></div></div> <div><div><div>GPA = 3.4</div><div>Kuliah yang diambil<ul style="list-style-type: none">PSP-4 sksSDA-3 sksMPKT-6 sks</div><div>Sangat Memuaskan!</div><div>Delete Data Update Data</div></div><div><div>No. 3</div><div>NPM = 125</div><div>Name = sav</div><div>GPA = 3.8</div><div>Kuliah yang diambil<ul style="list-style-type: none">PSP-4 sksDDP 1-4 sks</div><div>Cum Laude!</div><div>Delete Data Update Data</div></div></div>
--

STATIC FILE

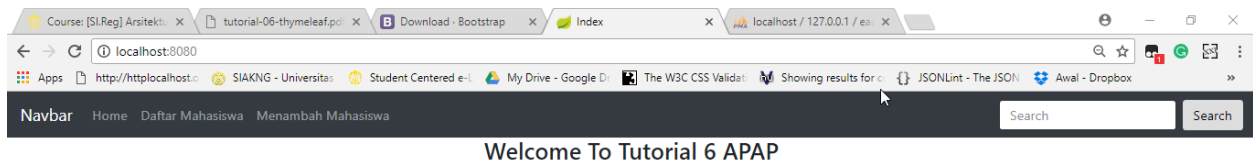
1. Unduh bootstrap pada <http://getbootstrap.com/docs/4.0/getting-started/download/>
2. Extract folder zip yang Anda unduh ke dalam folder static yang terdapat pada folder src/main/resources.



3. Buat halaman index.html menjadi source code dibawah ini

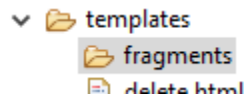
```
viewall.html StudentController.java ApapTutorial06Application.java index.html
1 <!DOCTYPE html>
2 <html xmlns:th="http://www.thymeleaf.org">
3   <head>
4     <title>Index</title>
5     <link rel="stylesheet" href="/css/bootstrap.min.css" />
6   </head>
7   <body>
8     <nav class="navbar navbar-expand-lg navbar-dark bg-dark">
9       <a class="navbar-brand" href="#">Navbar</a>
10      <button class="navbar-toggler" type="button" data-toggle="collapse"
11        data-target="#navbarSupportedContent"
12        aria-controls="navbarSupportedContent" aria-expanded="false"
13        aria-label="Toggle navigation">
14        <span class="navbar-toggler-icon"></span>
15      </button>
16      <div class="collapse navbar-collapse" id="navbarSupportedContent">
17        <ul class="navbar-nav mr-auto">
18          <li class="nav-item">
19            <a class="nav-link" href="/">Home</a>
20          </li>
21          <li class="nav-item">
22            <a class="nav-link" href="/student/viewall">Daftar Mahasiswa</a>
23          </li>
24          <li class="nav-item">
25            <a class="nav-link" href="/student/add">Menambah Mahasiswa</a>
26          </li>
27        </ul>
28      </div>
29    </nav>
30  </body>
31 </html>
```

4. Akses halaman index dari aplikasi Anda. Contoh tampilan adalah sebagai berikut.



FRAGMENT

1. Buat sebuah folder dalam folder templates dengan nama fragments



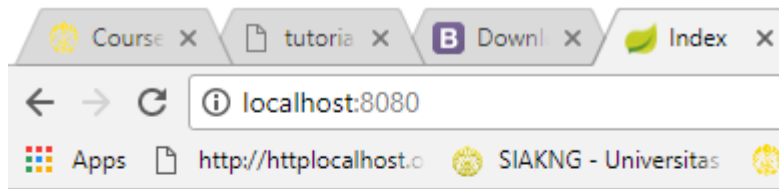
2. Buat sebuah halaman HTML bernama fragment.html yang berisi source code seperti dibawah ini. File HTML tersebut dibuat pada folder fragments yang baru Anda buat sebelumnya

```
fragment.html
1 <!DOCTYPE html>
2 <html xmlns:th="http://www.thymeleaf.org">
3 <head>
4   <title>Fragment</title>
5 </head>
6 <body>
7   <div th:fragment="header">
8     <ul>
9       <li><a href="/">Home</a></li>
10      <li><a href="/student/viewall">Daftar Mahasiswa</a></li>
11      <li><a href="/student/add">Menambah Mahasiswa</a></li>
12    </ul>
13  </div>
14  <div th:fragment="footer">
15    <hr/>
16    <h4>Mata Kuliah APAP</h4>
17  </div>
18 </body>
19 </html>
20
```

3. Pada source code diatas, th:fragment merepresentasikan nama fragment dari bagian yang Anda ingin jadikan sebuah fragment.
4. Buat halaman index.html menjadi source code dibawah ini

```
1 <!DOCTYPE html>
2 <html xmlns:th="http://www.thymeleaf.org">
3 <head>
4   <title>Index</title>
5 </head>
6 <body>
7   <div th:replace="fragments/fragment :: header"></div>
8   <h2>Welcome To Tutorial 6 APAP</h2>
9   <div th:replace="fragments/fragment :: footer"></div>
10 </body>
11 </html>
```

5. Coba jalankan halaman index yang Anda buat. Contoh tampilan yang keluar adalah sebagai berikut



- [Home](#)
- [Daftar Mahasiswa](#)
- [Menambah Mahasiswa](#)

Welcome To Tutorial 6 APAP

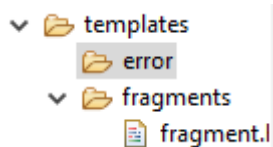
Mata Kuliah APAP

Pertanyaan 5 : Apa yang dimaksud dengan `th:replace="fragments/fragment :: header"` dan `th:replace="fragments/fragment :: footer"` pada file `index.html` yang Anda buat?

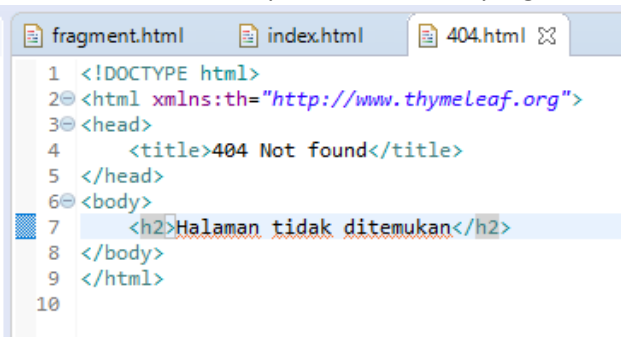
Jawab : yang dimaksud dengan `th:replace="fragments/fragment :: header"` dan `th:replace="fragments/fragment :: footer"` pada file `index.html` adalah sebagai penanda bahwa div yang berada di `fragment.html` dengan nama `header` dan `footer` yang ada di folder `fragments` akan diimport ke div `index.html` atau mengganti div di `index.html` dengan yang ada di `fragment.html`.

ERROR 404 NOT FOUND HANDLER

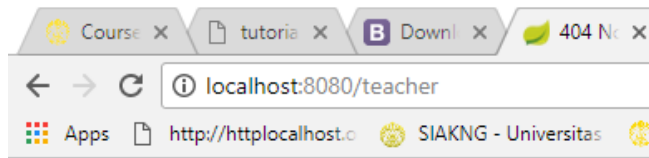
1. Buat sebuah folder dalam folder `templates` dengan nama `error`



2. Buat sebuah halaman HTML bernama `404.html` yang berisi source code seperti dibawah ini. File HTML tersebut dibuat pada folder `error` yang baru Anda buat sebelumnya



3. Akses suatu halaman yang tidak tersedia dari aplikasi Anda. Misalnya halaman dengan mapping /teacher. Contoh tampilan yang keluar adalah sebagai berikut



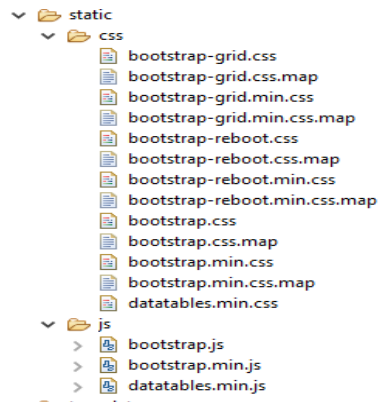
Halaman tidak ditemukan

Pertanyaan 6 : Apa handler dengan metode ini juga dapat berlaku bagi error lain seperti error 500: internal server error?

Jawab : error tersebut tidak terhandle dengan 404.html. error 500: internal server error dapat dihandle dengan membuat 500.html

LATIHAN

1. Pada tutorial, anda sudah mencoba untuk memperlihatkan apakah student tersebut cum laude atau tidak sampai ke bagian view. Pada latihan kali ini, anda diminta untuk mengubah view tersebut menggunakan DataTables. DataTables adalah plugin jquery library, yang dapat membuat sebuah tabel lebih interaktif dan lebih baik (<https://datatables.net/>). Oleh karena itu, gunakan data tabel untuk mengubah tampilan **localhost:8080/student/viewall**. Dengan kolom-kolom yang harus ada sebagai berikut:
 1. No
 2. NPM
 3. Name
 4. GPA
 5. Cum laude
 6. Delete (jika diklik, maka dapat menghapus student,)
 7. Update (jika diklik, yang dapat mengupdate student)Anda tidak harus menggunakan datables versi server side (https://datatables.net/examples/server_side/), yang mana lebih baik digunakan untuk data yang sangat banyak. Walaupun demikian, jika anda anda ingin mengimplementasi hal ini sangat diperbolehkan, tetapi tidak ada penambahan nilai untuk hal tersebut.
 - memasukkan css dan js datatables ke folder static



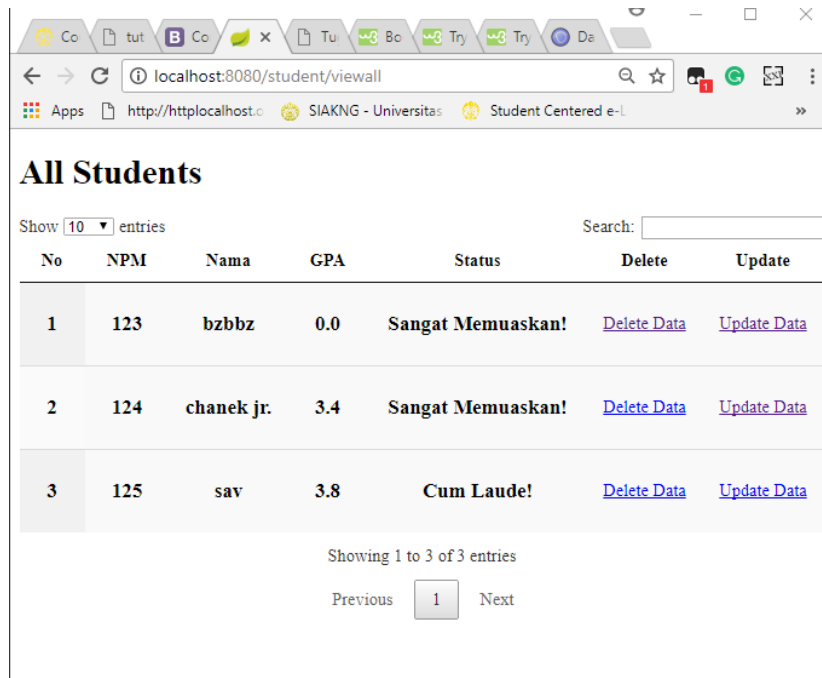
- memasukkan code dibawah ini di head viewall.html

```
<head>
  <title>View All Students</title>
  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.1/jquery.min.js"></script>
  <link rel="stylesheet" type="text/css" href="/css/datatables.min.css"/>
  <script type="text/javascript" src="/js/datatables.min.js"></script>
  <script>
    $(document).ready( function () {
      $('#viewall').DataTable();
    } );
  </script>
</head>
```

- Menambahkan tabel pada viewall.html

```
<body>
  <h1>All Students</h1>
  <table id="viewall" class="display" cellpadding="0" width="100%" style="text-align:center;">
    <thead>
      <tr>
        <th>No</th>
        <th>NPM</th>
        <th>Nama</th>
        <th>GPA</th>
        <th>Status</th>
        <th>Delete</th>
        <th>Update</th>
      </tr>
    </thead>
    <tbody>
      <tr th:each="student, iterationStatus: ${students}" th:class="${iterationStatus.odd}? 'odd'">
        <td><h3 th:text="${iterationStatus.count}">No. 1</h3></td>
        <td><h3 th:text="${student.npm}">Student NPM</h3></td>
        <td><h3 th:text="${student.name}">Student Name</h3></td>
        <td><h3 th:text="${student.gpa}">Student GPA</h3></td>
        <td><h3 th:text="${student.gpa}>3.49} ? 'Cum Laude!' : 'Sangat Memuaskan!'"></h3></td>
        <td><a th:href="/student/delete/" + ${student.npm}" >Delete Data</a></td>
        <td><a th:href="/student/update/" + ${student.npm}" >Update Data</a></td>
      </tr>
    </tbody>
  </table>
```

- localhost:8080/student/viewall



localhost:8080/student/viewall

All Students

Show entries Search:

No	NPM	Nama	GPA	Status	Delete	Update
1	123	bzbbz	0.0	Sangat Memuaskan!	Delete Data	Update Data
2	124	chanek jr.	3.4	Sangat Memuaskan!	Delete Data	Update Data
3	125	sav	3.8	Cum Laude!	Delete Data	Update Data

Showing 1 to 3 of 3 entries

Previous Next

2. Pada tutorial, Anda sudah mencoba membuat fragment header dan footer. Tetapi, header dan footer pada tutorial tersebut tidak dinamis. Data HTML pada header atau footer akan sama di setiap view yang memakai fragment tersebut.

Pada latihan kali ini, Anda diminta untuk membuat sebuah fragment header yang dinamis. Dimana, aspek dinamis tersebut terdapat pada title yang diletakkan pada fragment header, bukan setiap page HTML yang ada. Lebih jauh, title tersebut akan berubah-ubah sesuai dengan page yang ada. Silahkan implementasi dengan kebutuhan tersebut, dan jika sudah berhasil, pastikan header tersebut dimuat pada setiap view yang ada di project ini.

- membuat perubahan untuk title dan header di fragment.html

```
<!DOCTYPE html>
<html xmlns:th="http://www.thymeleaf.org">
<head>
<title th:fragment="title"
th:if="${#httpServletRequest.requestURI == '/'}">Index</title>
<title th:fragment="title"
th:if="${#httpServletRequest.requestURI == '/course/view/${id_course}'}">View
Course</title>
<title th:fragment="title"
th:if="${#httpServletRequest.requestURI == '/student/view/${npm}'}">View
Student by NPM</title>
<title th:fragment="title"
th:if="${#httpServletRequest.requestURI == '/student/viewall'}">View
All Students</title>
<title th:fragment="title"
th:if="${#httpServletRequest.requestURI == '/student/delete/${npm}'}">Delete
Student</title>
<title th:fragment="title"
th:if="${#httpServletRequest.requestURI == '/delete'}">Delete</title>
<title th:fragment="title"
th:if="${#httpServletRequest.requestURI == '/student/add'}">Add
Student</title>
<title th:fragment="title"
th:if="${#httpServletRequest.requestURI == '/student/add/submit'}">Add</title>
<title th:fragment="title"
th:if="${#httpServletRequest.requestURI == '/student/update/${npm}'}">Update
Student</title>
<title th:fragment="title"
th:if="${#httpServletRequest.requestURI == '/student/update/submit'}">Update</title>
</head>
```

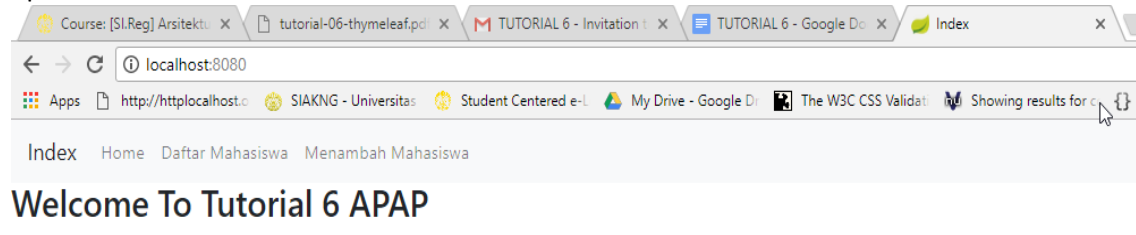
```
<body>
  <div th:fragment="header">
    <nav class="navbar navbar-expand-lg navbar-light bg-light">
      <a th:href="${#httpServletRequest.requestURI == '/'}"
        class="navbar-brand">Index</a> <a
        th:href="${#httpServletRequest.requestURI == '/course/view/${id_course}'}"
        class="navbar-brand">View Course</a> <a
        th:href="${#httpServletRequest.requestURI == '/student/view/${npm}'}"
        class="navbar-brand">View Student by NPM</a> <a
        th:href="${#httpServletRequest.requestURI == '/student/viewall'}"
        class="navbar-brand">View All Students</a> <a
        th:href="${#httpServletRequest.requestURI == '/student/delete/${npm}'}"
        class="navbar-brand">Delete Student</a> <a
        th:href="${#httpServletRequest.requestURI == '/delete'}"
        class="navbar-brand">Delete</a> <a
        th:href="${#httpServletRequest.requestURI == '/student/add'}"
        class="navbar-brand">Add Student</a> <a
        th:href="${#httpServletRequest.requestURI == '/student/add/submit'}"
        class="navbar-brand">Add</a> <a
        th:href="${#httpServletRequest.requestURI == '/student/update/${npm}'}"
        class="navbar-brand">Update Student</a> <a
        th:href="${#httpServletRequest.requestURI == '/student/update/submit'}"
        class="navbar-brand">Update</a>
      <button class="navbar-toggler" type="button" data-toggle="collapse"
        data-target="#navbarSupportedContent"
        aria-controls="navbarSupportedContent" aria-expanded="false"
        aria-label="Toggle navigation">
        <span class="navbar-toggler-icon"></span>
      </button>

      <div class="collapse navbar-collapse" id="navbarSupportedContent">
        <ul class="navbar-nav mr-auto">
          <li class="nav-item"><a class="nav-link" href="/">Home</a></li>
          <li class="nav-item"><a class="nav-link"
            href="/student/viewall">Daftar Mahasiswa</a></li>
          <li class="nav-item"><a class="nav-link" href="/student/add">Menambah
            Mahasiswa</a></li>
        </ul>
      </div>
    </nav>
  </div>
</body>
```

-pada setiap html tambahkan kode dibawah ini

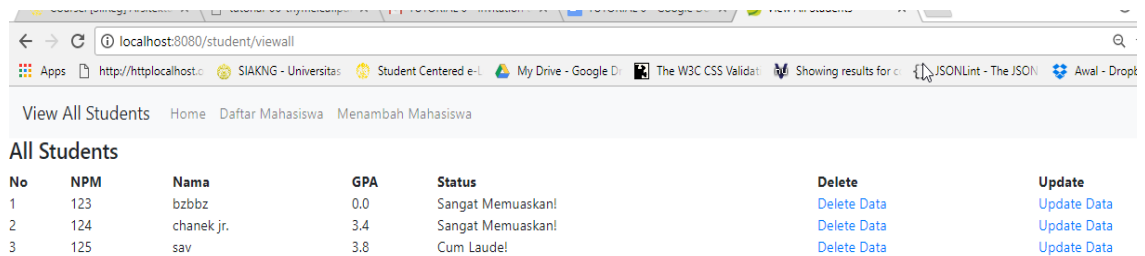
```
<!DOCTYPE html>
<html xmlns:th="http://www.thymeleaf.org">
<head>
  <title th:replace="fragments/fragment :: title"></title>
<script
  src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.1/jquery.min.js"></script>
<link rel="stylesheet" type="text/css" href="/css/datatables.min.css" />
<link rel="stylesheet" type="text/css" href="/css/bootstrap.min.css" />
<script type="text/javascript" src="/js/bootstrap.min.js"></script>
</head>
<body>
  <div th:replace="fragments/fragment :: header"></div>
</body>
```

- pada index



Mata Kuliah APAP

-pada viewall



No	NPM	Nama	GPA	Status	Delete	Update
1	123	bzbbz	0.0	Sangat Memuaskan!	Delete Data	Update Data
2	124	chanek jr.	3.4	Sangat Memuaskan!	Delete Data	Update Data
3	125	sav	3.8	Cum Laude!	Delete Data	Update Data

Dan hal itu terjadi pada html lainnya

Lesson Learned

Pada tutorial 6 saya mempelajari mengenai pemanfaatan thymeleaf untuk keperluan view. Thymeleaf dapat membantu dalam hal memanipulasi view agar sesuai dengan data yang ada. Selain itu, saya mempelajari bagaimana memasukan css dan js yang ada pada bootstrap atau framework css lainnya kedalam folder static. Sehingga kegunaan dari static adalah untuk menyimpan file css, js, dan lain-lain. Saya juga mempelajari mengenai bagaimana cara melooping suatu data pada view dan membedakan index ganjil dan genap.

Penggunaan conditional expression pada thymeleaf juga salah satu hal yang saya pelajari pada tutorial ini. Hal ini berguna untuk menampilkan sesuatu berdasarkan kondisi tertentu. Saya juga mempelajari mengenai fragment yang merupakan salah satu fitur thymeleaf dimana hal ini berfungsi agar kita dapat me-reuse code dari halaman html. Fragment ini berfungsi ketika kita menggunakan komponen-komponen yang sering muncul di html seperti contoh pada tutorial ini adalah pembuatan header dan footer.

Menghandle 404 Not Found Error juga merupakan hal yang saya pelajari jika kita ingin mengandle ketika user membuka page yang salah atau tidak ada pada program kita. Yang terakhir saya juga mempelajari membuat tabel yang interaktif menggunakan datatables yaitu plugin jquery library.